

**REMARKS**

Claims 1, 2, 4, 5, 8, 9, 11, 12, 15, 21, and 23 are amended to more clearly point out that which is being claimed.

Claims 6-7, 13-14, 17-20, 24 are cancelled without prejudice.

Claims 25-33 have been added to further point out that which is being claimed.

Claims 1-5, 8-12, 15, 16, 21-23, and 25-28 are pending.

In the specification, the "Related Applications" paragraph on page 1, beginning on line 2, has been amended to include the serial number, filing date and title of a related patent application having attorney docket number of MS1-710US.

**Rejections under 35 U.S.C. §103(a)**

Of the pending claims, Claims 1-4, 8-10, 15, 16, and 21-23 stand rejected under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent No. 5,649,196 issued to Woodhill et al. (*Woodhill*) and further in view of U.S. Patent No. 6,314,565 issued to Kenner et al. (*Kenner*). Applicants respectfully traverse these rejections for at least the following reasons.

*Woodhill* is directed towards distributed processing backup file systems and discloses that individual binary objects that are stored and backed-up periodically

1 can be assigned unique identifiers. *Kenner* discloses that script files can be used  
2 in a comparative way to determine which multimedia files to download.

3 **Claim 1** is directed towards a software version control method that includes  
4 assigning each of a plurality of data files to one of a plurality of specific  
5 corresponding downloadable file groups, and for each downloadable file group  
6 compressing together all assigned data files to form one processed image for the  
7 downloadable file group. The method also includes associating each resulting  
8 processed image with a unique identifier, generating a listing of unique identifiers,  
9 and storing the processed images and the listing of unique identifiers within a  
10 source device. The method also includes comparing the listing of unique  
11 identifiers with a current listing of unique identifiers in a client device, and  
12 selectively downloading processed images from the source device to the client  
13 device whose unique identifiers appear in the listing of unique identifiers but not  
14 in the current listing of unique identifiers in the client device.  
15

16  
17 Neither *Woodhill* nor *Kenner*, alone or together, disclose or suggest the  
18 software version control method in Claim 1. For example, the method includes  
19 assigning each of a plurality of data files to one of a plurality of specific  
20 corresponding downloadable file groups and that for each downloadable file group  
21 the method includes compressing together all assigned data files to form one  
22 processed image for the downloadable file group. *Woodhill* does not do this, but  
23 instead gives each binary object its own identifier. *Kenner* does not do this but  
24 rather deals only with specific individual software components.  
25

1       The method also includes, for example, associating each resulting  
2       processed image with a unique identifier, generating a listing of unique identifiers,  
3       storing the processed images and the listing of unique identifiers within a source  
4       device and comparing the listing of unique identifiers with a current listing of  
5       unique identifiers in a client device so as to selectively download processed  
6       images from the source device to the client device whose unique identifiers appear  
7       in the listing of unique identifiers but not in the current listing of unique identifiers  
8       in the client device. Since neither *Woodhill* and/or *Kenner* deal with such  
9       processed images, they also fail to disclose the ability for the software version  
10      control method to conduct a list comparison of unique identifiers for such  
11      processed images (i.e., compressed groups of related data files).  
12

13       In **Claim 2**, which depends from Claim 1, the source device is further  
14      specified to include at least one server device.  
15

16       **Claim 3**, which depends from Claim 1, further recites that each unique  
17      identifier is derived from its corresponding processed image.

18       **Claim 4**, which depends from Claim 1, further specifies that assigning data  
19      files to downloadable file groups includes assigning a plurality of related function  
20      data files to one downloadable file group.

21       **Claim 25**, which depends from Claim 1, further recites that the one  
22      processed image for the downloadable file group has a ".cim" extension.  
23

24       Independent **Claim 8** is directed towards a computer-readable medium  
25      having computer-executable instructions for causing at least one processing unit to

1 perform certain acts. The acts include assigning each of a plurality of data files to  
2 one of a plurality of specific corresponding downloadable file groups and for each  
3 downloadable file group compressing together all assigned data files to form one  
4 processed image for the downloadable file group. Further acts recited include  
5 associating each resulting processed image with a unique identifier, generating a  
6 listing of unique identifiers, and storing the processed images and the listing of  
7 unique identifiers within a source device. Additional acts include comparing the  
8 listing of unique identifiers with a current listing of unique identifiers in a client  
9 device, and selectively downloading processed images from the source device to  
10 the client device whose unique identifiers appear in the listing of unique identifiers  
11 but not in the current listing of unique identifiers in the client device.  
12

13 The exemplary arguments stated above with regard to Claim 1 are also  
14 applied to Claim 8.

15 **Claim 9**, which depends from Claim 8, further recites that the source  
16 device includes at least one server device.  
17

18 In **Claim 10**, which depends from Claim 8, it is further specified that each  
19 unique identifier is derived from its corresponding processed image.

20 **Claim 26**, which depends from Claim 8, further recites that the one  
21 processed image for the downloadable file group has a ".cim" extension.  
22

23 Independent **Claim 15** is directed towards an apparatus that includes  
24 memory and logic. The logic is operatively configured to assign each of a  
25 plurality of data files to one of a plurality of specific corresponding downloadable

1 file groups, and for each downloadable file group compress together all assigned  
2 data files to form one processed image for the downloadable file group. The logic  
3 also associates each resulting processed image with a unique identifier and stores  
4 the processed images and a listing of unique identifiers to the memory. The logic  
5 is also configured to compare the listing of unique identifiers with a current listing  
6 of unique identifiers in a client device to identify processed images that need to be  
7 provided to the client device.

8  
9 The exemplary arguments stated above with regard to Claim 1 are also  
10 applied to Claim 15.

11 **Claim 16**, which depends from Claim 15, further recites that each unique  
12 identifier is derived from its corresponding processed image.

13 **Claim 27**, which depends from Claim 15, further recites that the one  
14 processed image for the downloadable file group has a ".cim" extension.

15  
16 Independent **Claim 21** is directed towards a system that includes a network,  
17 a server device and a client device. The server device is configured to assign each  
18 of a plurality of server-based data files to one of a plurality of specific  
19 corresponding server-based downloadable file groups, and for each server-based  
20 downloadable file group compress together all assigned data files to form one  
21 processed image for the server-based downloadable file group. The server device  
22 is also configured to associate each resulting processed image with a unique  
23 identifier. The server device can also selectively output the processed images and  
24 a latest listing of unique identifiers over the network. The client device, which  
25

1 communicates with the server device through the network, is configured to  
2 maintain a listing of unique identifiers associated with processed images stored  
3 locally within the client device and to compare the listing of unique identifiers  
4 with a downloaded latest listing of unique identifiers from the server device, and  
5 selectively download processed images whose unique identifiers appears in the  
6 latest listing of unique identifiers from the server device but not in the listing of  
7 unique identifiers in client device.

8  
9 The exemplary arguments stated above with regard to Claim 1 are also  
10 applied to Claim 21.

11 **Claim 22**, which depends from Claim 21, further specifies that each unique  
12 identifier is derived from its corresponding processed image.

13 In **Claim 23**, which also depends from Claim 21, the server device recited  
14 as being further configured to selectively assign a plurality of related function data  
15 files to one downloadable file group.

16  
17 **Claim 28**, which depends from Claim 25, recites that the one processed  
18 image for the server-based downloadable file group has a “.cim” extension.

19 Of the pending claims, **Claims 5 and 12** stand rejected under 35 U.S.C.  
20 §103(a) as being unpatentable over *Woodhill* in view of *Kenner* and further in  
21 view of U.S. Patent No. 5,848,565 issued to Cowan (*Cowan*). Applicants  
22 respectfully traverse these rejections for at least the following reasons.

23  
24 *Cowan* is added to *Woodhill* and/or *Kenner* to add persistent memory to the  
25 mix.

1 Cowan does not, however, add the missing elements/features as recited in  
2 independent Claims 1 and 8, from which Claims 5 and 12, respectively, depend.

3 More specifically, **Claim 5** is directed towards a software version control  
4 method that includes assigning each of a plurality of data files to one of a plurality  
5 of specific corresponding downloadable file groups, and for each downloadable  
6 file group compressing together all assigned data files to form one processed  
7 image for the downloadable file group. The method also includes associating each  
8 resulting processed image with a unique identifier, generating a listing of unique  
9 identifiers, and storing the processed images and the listing of unique identifiers  
10 within a source device. The method also includes comparing the listing of unique  
11 identifiers with a current listing of unique identifiers in a client device, and  
12 selectively downloading processed images from the source device to the client  
13 device whose unique identifiers appear in the listing of unique identifiers but not  
14 in the current listing of unique identifiers in the client device. The method further  
15 includes sending the processed image and the listing of unique identifiers to a  
16 client device that stores the processed image and the listing of unique identifiers in  
17 a persistent memory  
18  
19

20 Neither *Woodhill*, *Kenner*, or *Cowan*, alone or combined, disclose or  
21 suggest the software version control method in Claim 5. The arguments presented  
22 above with respect to Claim 1 apply here too, for example.  
23

24 **Claim 12** is directed towards a computer-readable medium having  
25 computer-executable instructions for causing at least one processing unit to

1 perform certain acts. The acts include assigning each of a plurality of data files to  
2 one of a plurality of specific corresponding downloadable file groups and for each  
3 downloadable file group compressing together all assigned data files to form one  
4 processed image for the downloadable file group. Further acts recited include  
5 associating each resulting processed image with a unique identifier, generating a  
6 listing of unique identifiers, and storing the processed images and the listing of  
7 unique identifiers within a source device. Additional acts include comparing the  
8 listing of unique identifiers with a current listing of unique identifiers in a client  
9 device, and selectively downloading processed images from the source device to  
10 the client device whose unique identifiers appear in the listing of unique identifiers  
11 but not in the current listing of unique identifiers in the client device. Additional  
12 acts include sending the processed image and the listing of unique identifiers to a  
13 client device that stores the processed image and the listing of unique identifiers in  
14 a persistent memory.

15 Neither *Woodhill, Kenner, or Cowan*, alone or combined, disclose or  
16 suggest the computer-readable medium as recited in Claim 5. The arguments  
17 presented above with respect to Claims 1 and 8 apply here too, for example.

18 Independent **Claim 29** has been added, along with dependent **claims 30-33**.  
19 Claim 29 is directed towards a computer-readable medium having computer-  
20 executable instructions for causing at least one processing unit to perform acts that  
21 include assigning each of a plurality of data files to one of a plurality of specific  
22 corresponding downloadable file groups, for each downloadable file group,  
23 compressing together all assigned data files to form one processed image for the  
24 downloadable file group, associating each resulting processed image with a unique  
25



1 identifier, generating a listing of unique identifiers, and storing the processed  
2 images and the listing of unique identifiers within a source device.

3 The cited art neither discloses nor reasonably suggests this novel computer-  
4 readable medium, or those in Claims 30-33. The arguments presented above with  
5 respect to Claims 1 and 8, for example, apply here too.

6  
7 **Conclusion**

8 For at least the reasons presented above, the pending claims are clearly  
9 patentable over the cited art. It is respectfully requested, therefore, that the  
10 rejections be reconsidered and withdrawn and the application be allowed.

11  
12  
13 Respectfully Submitted,

14 Dated: 6/12/2003

15 By: \_\_\_\_\_

16 Thomas A. Jolly  
17 Reg. No. 39,241  
18 (509) 324-9256  
19  
20  
21  
22  
23  
24  
25